

Aplikativni protokoli

Aplikativni sloj je sedmi, odnosno najviši sloj OSI modela. Sva direktna manipulacija podataka i informacija se odvija na ovom nivou i time se omogućava pristup softveru i korisniku. Neki od servisa koje ovaj sloj pruža su: elektronska pošta, slanje i prijem fajlova, mrežni resursi, distribucija podataka korisniku itd.

Ovaj sloj se sastoji od različitih vrsta protokola. Jedan od njih je široko rasprostranjeni HTTP (HyperText Transfer Protocol), koji je osnova svetske računarske mreže (World Wide Web). Kada god u našem pretraživaču otvaramo stranu – pretraživač serveru, koji koristi HTTP, šalje ime željene strane i njenu lokaciju i server je šalje nazad.

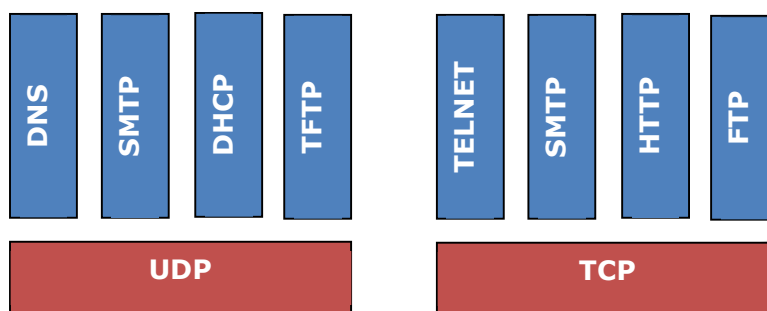
Napomena

World Wide Web i HTTP su usko povezani. Naime, World Wide Web predstavlja informacijski sistem gde se dokumenta i ostali resursi adresiraju uz pomoć URL-ova (Uniform Resource Locators), a dobivljaju i šalju preko HTTP-a. Dakle, World Wide Web nije sam internet, već samo njegov deo, iako se ta dva pojma često poistovećuju.

Drugi aplikativni protokoli su:

- FTP (File Transfer Protocol) – omogućava korisniku slanje fajlova i povlačenje fajlova sa udaljene mašine. Koristi TCP protokol. Prilikom povezivanja sa udaljenim uređajem, ostvaruju se dve konekcije: data konekcija i kontrolna konekcija. Data konekcija koristi port 20, a kontrolna konekcija port 21. Jedna konekcija (data) koristi se isključivo za transfer podataka, dok druga (kontrolna) služi za slanje komandi i pribavljanje odgovora.
- SMTP (Simple Mail Transfer Protocol) – omogućava korisniku slanje i primanje elektronske pošte. Koristi TCP protokol. SMTP protokol ima dve strane – klijentsku i serversku i koristi port 25.
- TELNET (remote login) – udaljeni pristup koji omogućava korisniku da sa jedne mašine šalje komande drugoj mašini – posredstvom terminala. Koristi TCP protokol. Ovo čini pomoću NVT (Network Virtual Terminal) sistema, koji kodira poslate komande na klijentskoj strani i dekodira ih na serverskoj strani. Port koji koristi je 23.
- DNS (Domain Name System) – ovaj sistem nam omogućava mapiranje IP adresa sa imenima sajtova na koje te adrese pokazuju. DNS je podeljen na tri kategorije:
 - Generički domen – ovaj domen se odnosi na „ponašanje” sajta – da li je reč o sajtu vlade države (.gov), da li je reč o sajtu organizacije (.org) ili pak edukativne ustanove (.edu). U ovom domenu se koriste tri karaktera.
 - Domen zemlje – ovaj domen predstavlja dvoslovnu skraćenicu zemlje kojoj sajt pripada (.me, .rs, .us). U ovom domenu se koriste dva karaktera.
 - Inverzni domen – ovaj domen se koristi da bi se IP adresa mapirala na ime sajta.

Neki od ovih protokola koriste UDP, a neki TCP kao transportni protokol. Podela aplikativnih protokola s obzirom na to koje transportne protokole koriste se može videti na sledećoj slici:



Slika 3.1. Podela aplikativnih protokola prema transportnim protokolima na kojima su bazirani

Funkcije aplikativnog sloja

- elektronska pošta – ovaj sloj pruža osnovu za slanje, primanje i skladištenje elektronske pošte (e-mail);
- mrežni virtualni terminali – omogućavaju korisniku povezivanje sa udaljenim uređajem; aplikacija emulira terminal na tom udaljenom uređaju, pa se tako ostvaruje komunikacija između korisnika i tog krajnjeg uređaja posredstvom terminala;
- prenos fajlova – korisnik ima mogućnost da pristupi udaljenom fajlu i rukuje njime, a može ga i dobiti;
- procesi na oba kraja (početni i destinacioni uređaji) poslednji su u lancu komunikacije.

Aplikativni protokol definiše kako procesi na datom uređaju (ili dva uređaja) prenose informacije između sebe. Pa tako, protokol aplikativnog sloja definiše:

- tip poruke – da li je reč o porukama kao zahtevima ili porukama kao odgovorima;
- sintaksu poruka – koja polja poruka sadrži i kako su ta polja u poruci razgraničena;
- značenje polja – značenje i odabir informacije koju će to polje sadržati;
- pravila i uslovi – kako i kada će proces slati poruke, odnosno odgovarati na poruke.

HTTP Protokol

HyperTextTransfer Protocol ili HTTP je protokol koji omogućava konekciju između web servera i klijenta. HTTP je aplikativni protokol za distribuiranje informacija preko weba. Koristi port 80. Takođe, HTTP klijent može biti pregledač (browser) kao što je FireFox, Google Chrome i drugi. U World Wide Webu, informacija se servira korišćenjem HyperText Markup Language (HTML) fajlova, dok je lokacija te informacije definisana URL-om (Uniform Resource Locator).

HTTP je prvi put definisan 1990. godine i prošao je kroz pet revizija:

- HTTP 0.9 – prvobitna implementacija protokola, koja je omogućavala jedino dobavljanje stranice;
- HTTP 1.0 – implementiran 1996. godine; u ovoj verziji se prvi put vide zaglavlja – tačnije dodatna polja prilikom zahteva;
- HTTP 1.1 – implementiran i ratifikovan 1997. godine; jedna od najbitnijih promena bila je trajna veza koja se nije završavala nakon prvog zahteva – dakle, omogućeno je više zahteva po jednoj konekciji;
- HTTP 2 – ratifikovan 2015. godine; sa ovom verzijom su zahtevi u okviru jedne konekcije postali paralelizovani, dok su zaglavlja kompresovana;
- HTTP 3 – predstavljen kao naslednik HTTP 2, prvi put se pojavljuje 2019. godine; najveća razlika je to što u ovom slučaju HTTP koristi UDP protokol.

Dobavljanje stranice

Web stranica je sačinjena od mnoštva različitih elementa i objekata – od HTML šablona od kojeg je sačinjena strana do slika i video-klipova. Proces dobavljanja svih ovih elemenata izgleda ovako:

- klijent šalje zahtev za željenom stranom serveru;
- server analizira zahtev i šalje nazad potvrdu klijentu zajedno sa HTML kodom koji čini stranu;
- klijent će početi sa analizom tog HTML koda i kreiranjem stranice (njegovim prikazom);
- dalje, klijent će u narednim zahtevima dobiti i sve ostale objekte, kao što su slike, video-klipovi i drugi multimedijalni izvori.

Nakon što su svi elementi dobavljeni, internet pretraživač će završiti prikazivanje stranice korisniku.

HTTP Metode

Pored mogućnosti da dobavimo podatke sa servera, HTTP omogućava i slanje podataka na server. Ovakvi mehanizmi slanja i primanja podataka u okviru HTTP-u su prikazani kao metode. Lista metoda koje HTTP podržava prikazana je u sledećoj tabeli:

METODA	OPIS
GET	metoda kojom dobavljamo željene podatke (dobavljanje stranice)
HEAD	metoda slična GET zahtevu, ali server mora vratiti samo HTTP zaglavlje
POST	metoda kojom klijent podnosi informacije ka serveru (slanje log-in forme)
PUT	metoda kojom klijent ostavlja podatak na serveru na zadatoj lokaciji
DELETE	metoda kojom klijent može obrisati podatak sa servera
TRACE	metoda koja omogućava klijentu da vidi zahtev koji je poslao serveru
OPTIONS	metoda koja prikazuje klijentu raspoložive opcije komunikacije sa serverom

Tabela 3.1. Tabelarni pregled HTTP metoda

Kada je reč o pretraživanju interneta, najkorišćenije metode su GET i POST. GET metoda se koristi za pribavljanje objekata i elemenata sa stranice (i same stranice), dok se POST metoda koristi u situacijama kada se traže određene informacije od klijenta (korisnika) – logovanje, internet kupovina itd.

Jedinstvena lokacija adrese HTTP-a (HTTP URL)

URL (jedinstvena lokacija adrese, engl. Uniform Resource Locator) – sadrži najbitniji podatak u GET zahtevu – a to je kombinacija lokacije gde se sajt nalazi, šeme pomoću koje se traži stranica sa tog sajta i putanja do imena fajla koji sadrži traženu stranu. Opciono, URL može sadržati podatke o broju porta ili parametre upita.

Takođe, postoji i URI (Uniform Resource Identifier), koji identifikuje stranicu po lokaciji, imenu ili po oba kriterijuma zajedno.

- <https://www.example.com/test.html> – URL;
- test.html – URI;
- www.example.com – URI.

Pitanje

TELNET:

- omogućava slanje fajlova
- **omogućava udaljeni pristup**
- omogućava slanje mejlova

Objašnjenje:

Tačan odgovor je da je TELNET protokol koji omogućava udaljeni pristup računaru (remote login).

Zaglavlja HTTP-a

Kako u HTTP-u postoje dve vrste poruka – zahtevi i odgovori, tako postoje i dva seta različitih zaglavlja koji se koriste.

HTTP poruka zahteva (HTTP request)

Primer na kome ćemo videti neke od uobičajenih polja zaglavlja koja se koriste prilikom slanja zahteva izgleda ovako:

Primer

```
GET /page.html HTTP/1.1

Connection: close

User-agent: Chrome/42.0.2311.135

Accept: text/html, image/jpeg

Accept-language:en
```

- GET /page.html HTTP/1.1 – ovom linijom u poruci nagoveštavamo koju HTTP metodu koristimo pri zahtevu (GET), koju stranicu sa servera želimo (/page.html), kao i verziju HTTP protokola koju koristimo (HTTP/1.1).
- Connection: close – Ovom linijom govorimo serveru da nakon tog, trenutnog, zahteva ne želimo da pravimo naredne zahteve i time okončavamo konekciju. Ako želimo da napravimo više zahteva u okviru jedne konekcije, koristićemo vrednost Keep-Alive, Connection zaglavlja zahteva.
- User-agent: Chrome/42.0.2311.135 – Zaglavljem User-agent nagoveštavamo kojim korisničkim agentom ćemo prikazati stranicu (tip pregledača koji koristimo). U ovom slučaju je to Chrome. Ovo polje je jako korisno jer omogućava serveru da prilagodi elemente na stranici u zavisnosti od toga šta taj pregledač podržava.
- Accept: text/html, image/jpeg – Ovim zaglavljem govorimo serveru koji tip podataka želimo da prihvatamo. U ovom slučaju su to tekstualni podaci – HTML stranica kao i slike, tačnije *.jpeg ekstenzija.
- Accept-language:en – Ovim zaglavljem nagoveštavamo da naš pretraživač preferira engleski jezik i da kao takav želi objekte prilagođene tom jeziku. Ako objekti tog jezika ne postoje na serveru, server će poslati objekte u podrazumevanom jeziku.

HTTP poruka odgovora (HTTP response)

Primer na kome ćemo videti neke od uobičajenih polja zaglavlja koja se koriste prilikom dopremanja odgovora izgleda ovako:

Primer

```
HTTP/1.1 200 OK

Connection: close

Date: Thu, 20 Aug 2020 12:00:00 GMT

Server: Apache/2.4.1 (Unix)

Last-Modified: Mon, 22 Jun 2020 09:21:42 GMT

Content-Length: 6821

Content-Type: text/html

'data ...'
```

Odgovor servera se može podeliti na tri sekcije: status, linije zaglavlja, i telo poruke – odnosno željeni podaci (linija: 'data...').

- `HTTP/1.1 200 OK` – Ovo polje u zaglavlju predstavlja statusnu poruku zahteva. Sastoji se od tri polja, verzije protokola (`HTTP/1.1`), statusnog koda (`200`) i statusne poruke na koju taj kod ukazuje (`OK`). Postoji više statusnih poruka; one se mogu pročitati [ovde](#).
- `Connection: close` – Označava mogućnost trenutnog zatvaranja konekcije ili, ako je prosleđen parametar `Keep-Alive` – ostavljamo mogućnost za podnošenje narednih zahteva serveru, a sve to u okviru iste konekcije.
- `Date: Thu, 20 Aug 2020 12:00:00 GMT` – Ovo polje označava datum i vreme (sati, minuti i sekunde) kada je ovaj odgovor generisan i poslat.
- `Server: Apache/2.4.1 (Unix)` – Ovo polje zaglavlja označava tip i verziju servera koji je generisao odgovor. Ovo polje je takođe analogno polju zaglavlja HTTP odgovora `User-Agent`, koje označava ime pretraživača koji šalje zahtev.
- `Last-Modified: Mon, 22 Jun 2020 09:21:42 GMT` – Označava datum i vreme kada je objekat (za koji je pretraživač poslao zahtev) poslednji put menjan ili kreiran. Ovo polje je veoma bitno kada je reč o lokalnom i serverskom keširanju podataka.
- `Content-Length: 6821` – Ukazuje na veličinu generisanog odgovora u bajtovima.
- `Content-Type: text/html` – Ovo polje pokazuje na tip tela poruke koje se šalje u odgovoru.

Napomena

Keširanje (engl. caching) je tehnika koja smešta kopiju datog resursa (objekta na stranici, same stranice) za ponovno korišćenje. U našem slučaju, ako bismo imali keširanu stranicu na našoj mašini, dakle stranicu koju smo već prethodno zatražili, umesto slanja zahteva ka serveru, pretraživač će iskoristiti upravo tu snimljenu kopiju. Ovim se rasterećuju serveri i ubrzava samo pretraživanje na mreži. Takođe, ovakva funkcionalnost je moguća i na serveru, ali da bi se iskoristila u potpunosti, mora biti pravilno konfigurisana.

Rezime

- Aplikativni sloj je najviši sloj OSI modela.
- Najkorišćeniji protokol aplikativnog sloja je HTTP, koji je baziran na TCP transportnom protokolu.
- Aplikativni protokol definiše tip poruke, sintaksu poruka, značenje polja i pravila i uslove prilikom komunikacije.
- URL (jedinствена lokacija adrese, engl. Uniform Resource Locator) – sadrži najbitniji podatak u GET zahtevu – a to je kombinacija lokacije gde se sajt nalazi, šeme pomoću koje se traži stranica sa tog sajta i putanje do imena fajla koji sadrži traženu stranu.
- HTTP funkcioniše po principu zahtev–odgovor.
- Setovi HTTP zaglavlja se biraju u odnosu na to da li se šalje zahtev ili dobija odgovor od servera.